WEST Search History

Hide Items | Restore | Clear | Cancel

DATE: Tuesday, January 03, 2006

Hide?	Set Name	Query	Hit Count
	DB=PGPB,USPT,USOC,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=ADJ		
☐	L5	L4 and @AD<2002012017	22
☐	L4	L3 and (policy or rule)	22
☐	L3	L2 and backup	23
☐	L2	L1 and sharing	83
☐	L1	content management service	159

END OF SEARCH HISTORY

Previous Doc Next Doc Go to Doc#
First Hit

Generate Collection

L5: Entry 19 of 22 File: PGPB Jun 27, 2002

DOCUMENT-IDENTIFIER: US 20020083118 A1
TITLE: Method and apparatus for managing a plurality of servers in a content delivery network

Application Filing Date:
20010518

Cross Reference to Related Applications Paragraph:
[0001] This application is a divisional of U.S. application Ser. No. 09/681,644, filed on May 15, 2001, entitled "Method and Apparatus For Large Payload Distribution in a Network," which claims the benefit of U.S. Provisional Application Ser. No. 60/266,286, filed on Oct. 26, 2000, entitled "Large Payload Delivery Networks Having Integrated Content Management Services," the specification of which is herein incorporated by reference.

Summary of Invention Paragraph:
[0025] Adding more storage requires rack space for mounting the new storage devices. Rack space is usually limited and sometimes expensive. Moreover, as storage capacity increases, more system administration functions (e.g., backup) are needed to manage the configuration. Since cost of system administration is expensive and rack space is limited, mirroring suffers.

Summary of Invention Paragraph:
[0039] A distributed file system is one in which files may be located on multiple servers connected over a local or wide area network. A distributed file system can be implemented using any one of several well-known network file system protocols, e.g., the Common Internet File System (CIFS) and Sun Microsystems, Inc.'s Network File System (NFS) protocol. CIFS is based on the standard Server Message Block (SMB) protocol widely in use by personal computers and workstations running a wide variety of operating systems. The CIFS protocol supports a number of file sharing and representation features, such as: file access, file and record locking, safe caching, read-ahead, and write-behind, file change notification, protocol version negotiation, extended attributes, distributed replicated virtual volumes, and server name resolution. NFS, like CIFS, is intended to provide an open cross-platform mechanism for client systems to request file services from server systems over a network. The NFS protocol provides transparent remote access to shared files across networks because it is designed to be portable across different machines, operating systems, network architectures, and transport protocols. NFS' portability is achieved through the use of Remote Procedure Call primitives (RPC primitives) that are built on top of system implementations that use the External Data Representation standard (XDR). The RPC primitives provide an interface to remote services. A server supplies programs (e.g., NFS), each program including a set of procedures. The combination of a server's network address, a program number, and a procedure number specifies a specific remote procedure to be executed. XDR uses a language to describe data formats. The language can only be used to describe data; it is not a programming language. NFS Implementations exist for a wide variety of systems. NFS mount protocol allows the server to hand out remote access privileges to a restricted set of clients and to perform various operating system-specific functions that allow, for example, attaching a remote directory tree to a local

file systems.

Detail Description Paragraph:
[0126] From one perspective, each stored block in the system storage of an SCDN node corresponds to a contiguous segment of a large payload file (e.g., a contiguous interval of movie). For example, the segments that comprise a movie, if viewed one after the other from the first segment to the last segment, would result in viewing the entire movie. Since the same content portions (i.e., segments) are located at several different nodes in the SCDN, non-contiguous segments of a file (e.g., non-contiguous portions of a film) can be retrieved independently and in parallel. This has several important side effects. For example, since a DS can obtain needed content portions from several different distribution servers, the reliability and availability of the SCDN are significantly increased. Additionally, the end-user can efficiently access segments of a large payload "out-of-order", e.g., fast-forwarding of a movie can be realized without actually having to download all of the portions of the film that are not actually viewed. Importantly, pruning (freeing the storage used by some blocks for use by other blocks) can be done at the "block level" (versus the entire "file level") based on specific content provider policies, e.g., pruning can be based on usage patterns. Usage of the content can also be rated at the block level.

Detail Description Paragraph:
[0139] There are additional situations in which learning and adaptation processes may be used in other embodiments of the invention. For example, as a large payload file is accessed, VFCS serves the content to Application Servers (such as Streaming Servers), while it also communicates with distribution servers to pull missing content portions from other locations. As more and more content portions are downloaded to satisfy end-user requests, the storage space for each content provider must be carefully monitored. Based on storage availability and usage information collected by VFCS, a pruning process could be used to remove certain blocks of media files. The policy associated with the pruning process should address: (1) when to prune, (2) how much to prune, and (3) which blocks to prune. After pruning, a server's storage system may contain entire media files or non-contiguous segments of files that are accessed frequently by local users. Additionally, the content provider might be apprised that more storage or more Distribution Servers, Application Servers, or VFCS Servers should be added to the network.

Detail Description Paragraph:
[0197] In one embodiment of the invention, the Service Management Subsystem 2080 includes the following components: History Log Handler, Statistics Handler, Event Handler, Threshold Monitor, Trap Agent, SNMP (Simple Network Management Protocol) stack, Presentation Agent, and Service Agreement Policy Agents. A History Log Handler and a Statistics Handler collect statistics and task/transaction logs from the local devices and Servers, and save all log and statistic information into the History and Statistics Repository. While a station is learning, a History Log Handler forwards all the file download records to the Learning Agent to notify the agent of the download status. This handler also forwards the inbound and outbound data transfer information recorded from local DSs to the Content Provider Storage Usage Table. The Content Usage and Statistics Database is also updated by the Statistic Handler.

Detail Description Paragraph:
[0199] Data from the History and Statistics Repository and Event Repository can be either pushed to, or pulled from, a Presentation Agent. Service Agreement Policy Agents retrieve data from History and Statistics Repository and feed the information to a Service Agreement Policy Server, where business agreement and policy (such as guaranteed quality of service per customer) can be enforced.

Detail Description Paragraph:

[0216] The Content Provider Data Table includes information such as content provider account information, the content provider assigned Content Management Server address, reserved storage, content provider's policy server, etc.

Previous Doc Next Doc Go to Doc#

Previous Doc Next Doc Go to Doc#
First Hit

Generate Collection

L3: Entry 12 of 28 File: PGPB Nov 28, 2002

DOCUMENT-IDENTIFIER: US 20020178271 A1
TITLE: Dynamic file access control and management

Application Filing Date:
20011120

Summary of Invention Paragraph:
[0017] d. Flexibility--The proxy file management system includes the ability to flexibly represent the content management needs of the enterprise through policy. Thus, flexible and straightforward interfaces to content management systems is provided.

Detail Description Paragraph:
[0061] d. Flexibility--The proxy file management system includes the ability to flexibly represent the content management needs of the enterprise through policy. Thus, flexible and straightforward interfaces to content management systems is provided.

Detail Description Paragraph:
[0199] Files stored on the NAS storage system 160 are stored in the NAS-native file-system format in plain-text. However, encryption costs are mitigated by storing the encrypted images of recently accessed files in a proxy file management system 100 directory tree mounted off the root directory of the NAS file-system 160. This prevents the loss of data following a catastrophic failure and allows existing backup software to operate normally. Moreover, this limits the possible data loss caused by bugs in the content subsystem 220 or client module 230.

Detail Description Paragraph:
[0241] The present invention modifies files sent across networks in real-time. When a client machine requests a file from an HTTP server (such as Apache or IIS), the request is routed to an dynamic content management server, whereby the file is retrieved, encrypted (optionally), rules are applied, and then the requested file, along with the "other" information wrapped with the file, is sent to the requesting client. For example, if a musician wants to sell his/her music online, that person can choose the file format they prefer (e.g., MP3,WMA, ePAC, etc.) and simply instruct the dynamic content management server to securely distribute the file under a specified set of rules. This removes the trial and error associated with encryption and encoding in conventional systems.

Detail Description Paragraph:
[0243] When an end-user receives a file which has been "wrapped" by the DCMS, the user can open it with an application that is compatible with the end-user's native operating system, provided that the dynamic content management client software has been installed on the end- user's device. The end-user can only use the file in a manner defined by the rules that have been sent by the file's creator (or administrator). These rules can specify a variety of different variables, such as the number of playbacks (or openings of the file), whether the file can be transferred to portable devices, and whether the file will ever "expire" (e.g., file cannot be accessed after a particular day and/or time of day). These rules are

stored on (or accessed by) the dynamic content management server or servers, and are attached to the file ("wrapped") at the time of download, as part of the dynamic content management server process. Thus, the DCMS lends itself to all areas of secure media distribution where privacy, copyright, bandwidth management, additional revenue streams and/or financial protection are required.

Previous Doc Next Doc Go to Doc#